Jurnal 8
Khalifardy Miqdarsah
1304211035

**a. Output**

```
Padi
SemuaTakSama
Rock
25

OneOkRock
WhereverYouAre
Rock
23

FiersaBesari
Celengan
Indie
15

Letto
SebelumCahaya
Pop
28

Letto
PermintaanHati
Pop
20

Tipe-X
genit
Ska
22

Noah
WalauHabisTerang
Alternatif
30
--------------
LAGU PALING BANYAK DI PUTAR
--------------
WalauHabisTerang
Noah

List Kosong
```

**b. source code**

**CSLL.h**

```
#ifndef CSLL_H_INCLUDED
#define CSLL_H_INCLUDED

#include <iostream>
using namespace std;

#define nil NULL
#define info(P) (P)->info
#define next(P) (P)->next
#define first(L) ((L).first)

struct infotype {
    string artis;
    string judul;
    string genre;
    int playtime;
};

typedef struct element *address;

struct element {
    infotype info;
    address next;
};

struct ListLagu{
    address first;
};
```

```
void create_List(ListLagu &L);
void createElemen(infotype laguBaru, address &pLagu);
void insertFirst(ListLagu &L, address pLagu);
void insertLast(ListLagu &L, address pLagu);
void deleteFirst(ListLagu &L, address &pLagu);
void deleteLast(ListLagu &L, address &pLagu);
void showSemuaLagu(ListLagu L);
address createl(infotype laguBaru);

//Jurnal
void tambahLagu(ListLagu &L, address pLagu, string posisi);
void showMostPlay(ListLagu L); //1304211035
address cariLagu(ListLagu L, string artis, string judul);
void resetPlayList(ListLagu &L);

#endif // CSLL_H_INCLUDED
```

**CSLL.cpp**

```
#include "CSLL.h"

void create_List(ListLagu &L){

    first(L) = nil;
};


void createElemen(infotype laguBaru, address &pLagu){

    pLagu = new element;
    info(pLagu) = laguBaru;
    next(pLagu) = nil;
};


void insertFirst(ListLagu &L, address pLagu){

    address P;
    if (first(L)==nil){

        next(pLagu)=pLagu;
        first(L) = pLagu;
    }else{

        P = first(L);

        while (next(P) != first(L)){

            P = next(P);
        };

        next(P)= pLagu;
        next(pLagu) = first(L);
        first(L) = pLagu;



    };
};
```

```
void insertLast(ListLagu &L, address pLagu){
    address P;

    if (first(L)==nil){
        insertFirst(L,pLagu);
    }else{

        P = first(L);
        while (next(P) != first(L)){

            P = next(P);
        };

        next(pLagu) = first(L);
        next(P) = pLagu;

    };

};
```

```
void deleteFirst(ListLagu &L, address &pLagu){
    address P;

    if (first(L) != nil){
            P = first(L);
            pLagu = first(L);

            if (next(P) == first(L)){
                next(pLagu) = nil;
                first(L) = nil;
            }else{

                while (next(P) != first(L)){
                    P = next(P);

                };

                next(P) = next(first(L));
                next(pLagu) = nil;
                first(L) = next(first(L));


            };



    };


};
```

```
void deleteLast(ListLagu &L, address &pLagu){
    address P;

    if (first(L) != nil){
            P = first(L);
            pLagu = first(L);

            if (next(P) == first(L)){
                next(pLagu) = nil;
                first(L) = nil;
            }else{

                while (next(next(P)) != first(L)){
                    P = next(P);

                };

                pLagu = next(next(P));
                next(P) = first(L);
                next(pLagu) = nil;
            };

    };

};
```

```
void showSemuaLagu(ListLagu L){
    address P;

    if (first(L) != nil){

        P = first(L);
        while (next(P) != first(L)){
            cout<<info(P).artis<<endl;
            cout<<info(P).judul<<endl;
            cout<<info(P).genre<<endl;
            cout<<info(P).playtime<<endl;
            cout<<endl;

            P = next(P);
        };

        cout<<info(P).artis<<endl;
        cout<<info(P).judul<<endl;
        cout<<info(P).genre<<endl;
        cout<<info(P).playtime<<endl;
    }else{
        cout<<"List Kosong"<<endl;

    };

};
```

```
//Jurnal

void tambahLagu(ListLagu &L, address plagu, string posisi){

    if (posisi == "awal"){
        insertFirst(L,plagu);
    }else{
        insertLast(L,plagu);
    };
};
```

```
void showMostPlay(ListLagu L){

    address P,MAX;

    if (first(L) != nil){
        P = first(L);
        MAX = first(L);

        while (next(P) != first(L)){
            if (info(MAX).playtime<= info(P).playtime){
                MAX = P;
            };

            P = next(P);
        };

        if (info(MAX).playtime<= info(P).playtime){
                MAX = P;
            };

        cout<<info(MAX).judul<<endl;
        cout<<info(MAX).artis<<endl;
    }else{
        cout<<"List Kosong"<<endl;
    };
};
```

```
address cariLagu(ListLagu L, string artis, string judul){

    address P;
    bool tidak_ketemu;

    P = first(L);
    tidak_ketemu = true;

    while(next(P) != first(L)){

        if (info(P).artis == artis && info(P).judul == judul){
            tidak_ketemu = false;
            break;
        };

        P = next(P);
    };

    if (tidak_ketemu){
        if (info(P).artis==artis && info(P).judul==judul){
            return P;
        }else{
            return nil;
        };
    }else{
        return P;
    };
};
```

```
void resetPlayList(ListLagu &L){

    address pLagu;
    while(first(L)!=nil){
        deleteFirst(L,pLagu);
        pLagu = nil;
    };
};
```

**Main.cpp**

```
#include "CSLL.h"
int main()
{
    ListLagu L;
    address pLagu;
    string posisi,artis,judul,genre;
    int playtime, cnt;
    infotype data;

    create_List(L);
    cnt = 0;

while(cnt < 7){
    cout<<"Artis: ";
    cin>>artis;
    cout<<endl;
    data.artis = artis;

    cout<<"Judul: ";
    cin>>judul;
    cout<<endl;
    data.judul = judul;

    cout<<"Genre: ";
    cin>>genre;
    cout<<endl;
    data.genre = genre;

    cout<<"playtime: ";
    cin>>playtime;
    cout<<endl;
    data.playtime = playtime;

    cout<<"Posisi: ";
    cin>>posisi;
    cout<<endl;

    createElemen(data,pLagu);
    tambahLagu(L,pLagu,posisi);

    cnt+=1;
};
```

```
showSemuaLagu(L);
cout<<"--------------"<<endl;
cout<<"LAGU PALING BANYAK DI PUTAR"<<endl;
cout<<"--------------"<<endl;
showMostPlay(L);

resetPlayList(L);
cout<<endl;
showSemuaLagu(L);


return 0;
```